PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

INTRODUCTION :

In genetic diagnosis, accurate clinical judgment is dependent on the proper classification of germ-line variants. The germ-line variants are classified into three condensed categories i.e. benign or likely benign, Variant of Uncertain Significance(VUS), and likely pathogenic or pathogenic.

In some cases, different labs can wrongly classify the germ-line variant for the same gene. For instance, if a doctor orders a genetic test from lab A for patient A and they find a genetic variant with "Pathogenic" classification. The doctor does a literature search and finds that this variant is likely to contribute to the patient's disease and decide to make some risk-reducing decision for patient A. Now, say a different doctor orders a genetic test from lab B for patient B and they find the same genetic variant,however, lab B classifies this variant as "Benign". This doctor might choose to seek further testing for the patient or do nothing at all. Such a classification is called conflicting clinical classification and it may cause flawed medical management.The research question of this project is to predict whether a variant will have a conflicting clinical classification.

My proposition is to apply machine learning methods i.e. classification technique in predictive analysis to identify the conflicting variant by comparing different lab results and thereby predicting whether a genetic variant will have a conflicting clinical classification in future analyses.

LITERATURE REVIEW:

Previous researches regarding germ-line variant classification have stated that there is discordance among clinicians and researchers in accurately classifying the variant. The difference in predicting the accurate germ-line variant by different labs have caused clinical impact on patients and among family members.

During a research regarding the inheritance risk of death due to Cardiomyopathy, a deceased patient's family were also tested for the presence of the pathogenic germ-line variant of the aforementioned condition. The family members who were negatively tested were considered not at risk and the positively tested ones were instructed to be regularly monitored for emergence of the disease. In due course of time, a different lab with the help of updated determination techniques has reclassified the pathogenic variant of a positively tested family member to 'likely benign' variant. A renewed test also determined that some previously negatively tested family members are now found to be at risk [*N Engl J Med, 2015*]. In another genetic variant detection study, its has been reported that concordance rates differ among clinical areas and variant types [*Genet Med, 2017*].

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

To overcome such problems different methods have been introduced based on guidelines such as interpreting germ-line variant based on American College of Medical Genetics and Genomics (ACMG) and Association of Molecular Pathology (AMP) guidelines [*Genet Med, 2017*], scoring system was introduced by labs based on ACMG guidelines [*Colleen Caleshu and Euan A. Ashley, 2016*] , 'in silico' algorithm was introduced by ACMG pathologists which is based on ACMG and AMP guidelines [*Genome Biology, 2017*].

Besides ACMG guidelines different laboratories have developed their own methods of variant classification, as variant-reporting guidelines did not address the weighting of evidence for variant classification. As an alternative, application of the automated tool for calculation of overall classification from evidence code was developed which is a pathogenicity calculator used to compare calculated ACMG-AMP classification based on the evidence code provided by the labs with final ACMG-AMP classification submitted by labs [*LM Amendola, 2016*].

Aiming to improve the interpretation process based on ACMG-AMP guidelines, web-based tools and software systems were developed. Prospective Registry of Multiplex Testing (PROMPT) is a web-based platform to assess the cancer risk of genetic variants [*J Clin Oncol, 2016*], ClinGen Pathogenicity calculator [*Genomic Medicine, 2017*], CharGer (Characterization of germ-line variant) etc are other software tools used for interpreting and predicting clinical pathogenicity of germ-line variant [*Adam D Scoff, Kuan Ling Huang, 2018*].

All these methods and software tools have helped to reduce discordance among different labs in predicting accurate germ-line variant to some extent. By applying machine-learning to predict a conflicting clinical variant will help the labs to accurately classify the germ-line variant.

DATASET :

The data-set that is used for this analysis can be found at Kaggle (https://www.kaggle.com/ kevinarvai/clinvar-conflicting) and for further reference, raw ClinVar.vcf is used (ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh37/clinvar.vcf.gz). Removed all variants from the original ClinVar.vcf which only had one submission as the problem only relates to variants with multiple classifications

The data-set is comprised of 46 attributes and 65188 rows which consists of numerical as well as categorical values. It is represented as a binary classification problem. It is a binary representation of whether or not a variant has a conflicting clinical classification, here '0' represents consistent classifications and '1' represents conflicting classifications. These classifications has been assigned to the class column. The attributes that will be used for analysis is as follows: CHROM, POS, REF, ALT, AF_ESP Allele, AF_EXAC Allele, AF_TGP Allele, CLNVC, MC,

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

ORIGIN,CLASS, Allele, Consequence, IMPACT, SYMBOL, Feature_type, Feature Ensemble, BIOTYPE, EXON, INTRON, cDNA_position, CDS_position, Codons, Protein_position, Amino_acids, STRAND, PolyPhen, LoFtool, CADD_PHRED, CADD_RAW, SIFT, BLOSUM62, BAM_EDIT.

CHROM : Chromosome on which the variant is located .

POS : Position on the chromosome the variant is located on.

REF : Reference Allele

ALT : Alternaete Allele

AF_ESP : Allele frequencies from GO-ESP

AF_EXAC : Allele frequencies from ExAC

AF_TGP : Allele frequencies from the 1000 genomes project

CLNDISDB : Tag-value pairs of disease database name and identifier.

CLNDISDBINCL : For included Variant: Tag-value pairs of disease database name and identifier.

CLNDN : ClinVar's preferred disease name for the concept specified by disease identifiers in CLNDISDB

CLNDNINCL : ClinVar's preferred disease name for the concept specified by disease identifiers in CLNDISDB

CLNHGVS : Top-level (primary assembly, alt, or patch) HGVS expression.

CLNSIGINCL : Clinical significance for a haplotype or genotype that includes this variant. Reported as pairs of VariationID:clinical significance.

CLNVC : Variant Type

CLNVI : the variant's clinical sources reported as tag-value pairs of database and variant identifier

MC : comma separated list of molecular consequence in the form of Sequence Ontology ID| molecular_consequence

ORIGIN : Allele origin. One or more of the following values may be added: 0 - unknown; 1 - germline; 2 - somatic; 4 - inherited; 8 - paternal; 16 - maternal; 32 - de-novo; 64 - biparental; 128 - uniparental; 256 - not-tested; 512 - tested-inconclusive; 1073741824 - other

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

SSR : Variant Suspect Reason Codes. One or more of the following values may be added: 0 - unspecified, 1 - Paralog, 2 - byEST, 4 - oldAlign, 8 - Para_EST, 16 - 1kg_failed, 1024 - other

CLASS : The binary representation of the target class. 0 represents no conflicting submissions and 1 represents conflicting submissions.

Allele : The variant allele used to calculate the consequence

Consequence : Type of consequence.

IMPACT: the impact modifier for the consequence type

SYMBOL : Gene Name

Feature_type : type of feature. Currently one of Transcript, RegulatoryFeature, MotifFeature.

Feature : Ensemble stable ID of feature

BIOTYPE : Biotype of transcript or regulatory feature

EXON : the exon number (out of total number)

INTRON : the intron number (out of total number)

cDNA_position : relative position of base pair in cDNA sequence

CDS_position : relative position of base pair in coding sequence

Protein_position : relative position of amino acid in protein

Amino_acids : only given if the variant affects the protein-coding sequence

Codons : the alternative codons with the variant base in upper case

DISTANCE : Shortest distance from variant to transcript

STRAND : defined as + (forward) or - (reverse).

BAM_EDIT : Indicates success or failure of edit using BAM file

SIFT : the SIFT prediction and/or score, with both given as prediction(score)

PolyPhen : the PolyPhen prediction and/or score

MOTIF_NAME : the source and identifier of a transcription factor binding profile aligned at this position

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

MOTIF_POS : The relative position of the variation in the aligned TFBP

HIGH_INF_POS : a flag indicating if the variant falls in a high information position of a transcription factor binding profile (TFBP)

MOTIF_SCORE_CHANGE : The difference in motif score of the reference and variant sequences for the TFBP

LoFtool : Loss of Function tolerance score for loss of function variants

CADD_PHRED : Phred-scaled CADD score.

CADD_RAW : Score of the deleterious of variants

Descriptive statistics of the data-set and attributes is that there are missing values in the data-set. MOTIF_NAME, SSR, MOTIF_POS, HIGH_INF_POS, CLNDISDBINCL, CLNVI, MOTIF_SCORE_CHANGE, DISTANCE, CLNDISDB, CLNDNINCL ,BIOTYPE, EXON, INTRON, cDNA_position, CDS_position, Codons, Protein_position, Amino_acids, ORIGIN has missing values. The CLASS distribution is skewed a bit to the 0 class, which mean that there are fewer variants with conflicting submissions. As the mean of the target attribute (CLASS) is greater than that of the median and standard deviation is not equal to 1 it shows that our data-set is not normally distributed.

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

APPROACH :

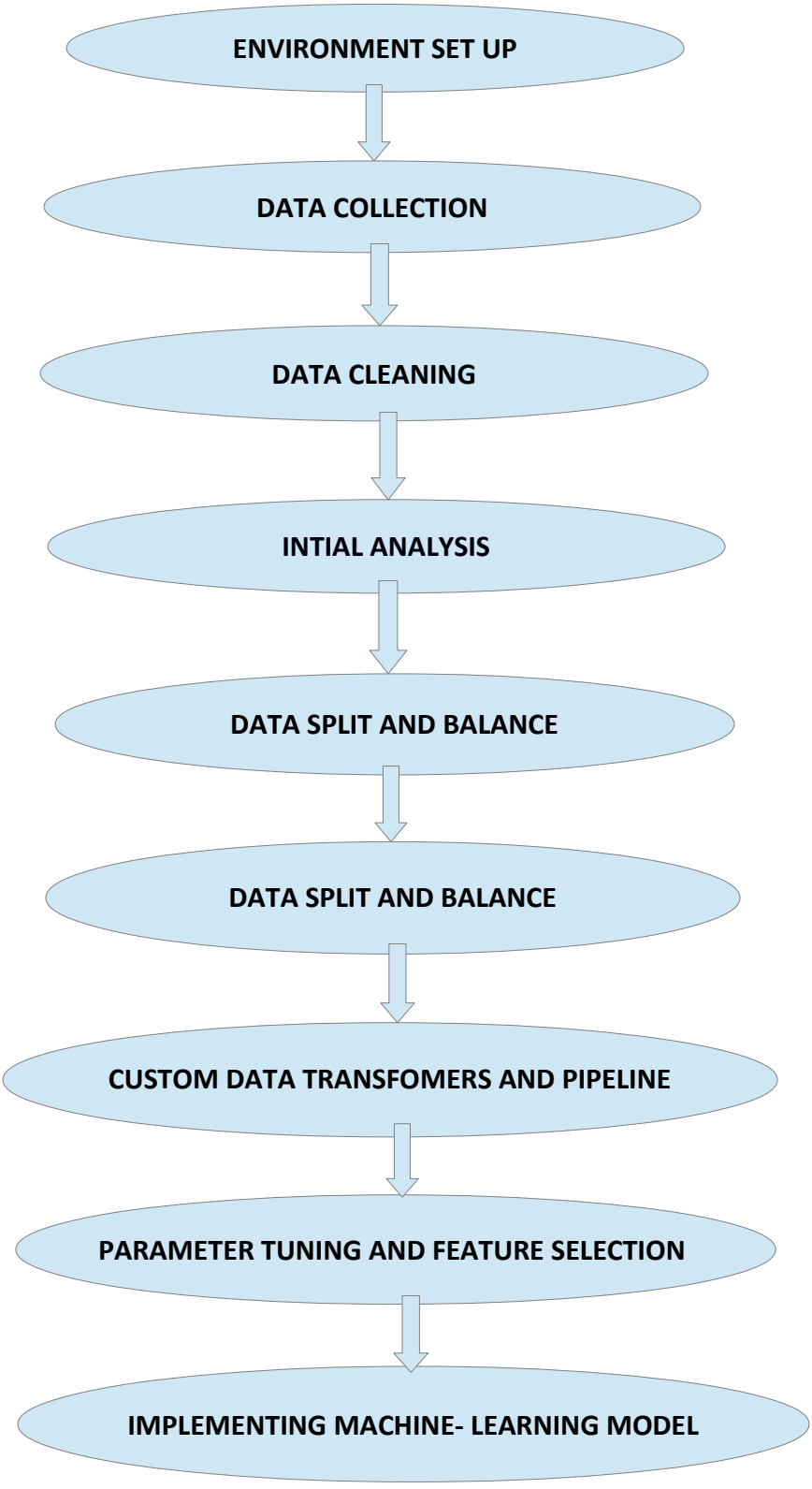


Figure 1:Step by step Approach

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

STEP 1 : ENVIRONMENT SET UP :

- Python environment has been setup on jupyter platform and all the necessary libraries which are essential for data processing has been installed through anaconda prompt .
- Imported libraries from numpy , pandas , scipy.stats as stats to processs the data for intial analysis.
- **Github Link :** https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

STEP 2 : DATA COLLECTION :

- Download the genetic variant dataset from Kaggle (https://www.kaggle.com/ kevinarvai/clinvar-conflicting) for data analysis .
- Import the .csv file into python kernel through panda dataframe for further processing .
- **Github Link :** https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

STEP 3 : DATA CLEANING :

- Removing unwanted observation and Missing Data includes missing values, irrelevant data and duplicate data. Missing values occurs when no value is stored for a certain observation within a variable. Duplicate observations most frequently arise during data collection which usually occurs when combing data-set from multiple places. Irrelevant observations are those that have least significant impact or one's that don't exactly fit the specific problem that you're trying to solve.
- In our data-set we can observe that variables such as 'MOTIF_NAME', 'CLNVI' 'MOTIF_POS','MOTIF_SCORE_CHANGE', 'CLNDISDB', 'HIGH_INF_POS', 'CLNDISDBINCL', 'CLNDNINCL', 'CLNSIGINCL' have around 100 percent missing value . After referring few research related to this domain ,came to a conclusion that 'DISTANCE', 'SSR' are least significant values in the data-set. Hence eliminating these attributes.
- **Github Link :** https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

STEP 4 : INTIAL ANALYSIS :

- In intial analysis we have to look into few basic questions such as :

  How many features does the datset have?

  What are the data types of my features?

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

Which is the target variable?

- Our dataset has 64 attributes and 65188 rows which contains a mix of numerical and categorical values.
- We need to verify datatype of each attribute and convert the datatype of each attribute to unique datatype.
- As we explore the data, we can find attribute such 'CHROM', 'REF', 'ALT', 'Allele', 'cDNA_position', 'CDS_position', 'Protein_position'. 'INTRON', 'EXON' ,'CLNDN' has numeric and categorical values in same attribute, some attribues have string format which is converted into unique datatype with help of fit and transform method in python.
- Further with help of sklearn.pipeline we imported Pipeline function to combine the transformed data and fit into our orignal dataset .
- Differentiate the attributes into numerical and categorical based on the datatype of each attribute.
- 'CLASS' attribute is our target variable which is binary representation of 1 and 0. We have 48754 0's and 16434 1's in our target attribute.
- **Github Link** : https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

STEP 4: EXPLORATORY DATA ANALYSIS:

- **Plot Categorical Distributions :** For plotting graphs in python we have used import seaborn as sns function and pandas cross-tab method to plot categorical variables 'CHROM', 'IMPACT', 'STRAND', 'BAM_EDIT', 'SIFT', 'PolyPhen','BLOSUM62', 'Consequence', 'CLNVC' against target variable class . As categorical features cannot be visualized through histograms we have used differnt type of bar plots.

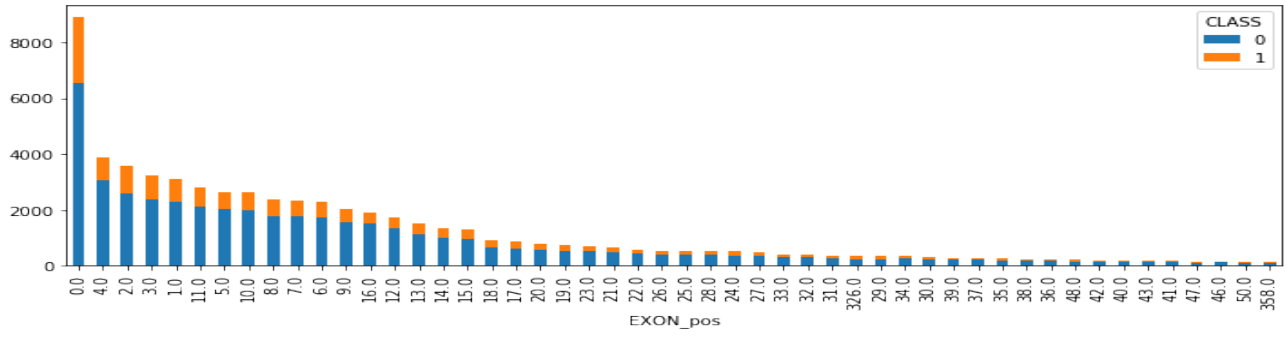


Figure 2: Bar plot of Exon_pos

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

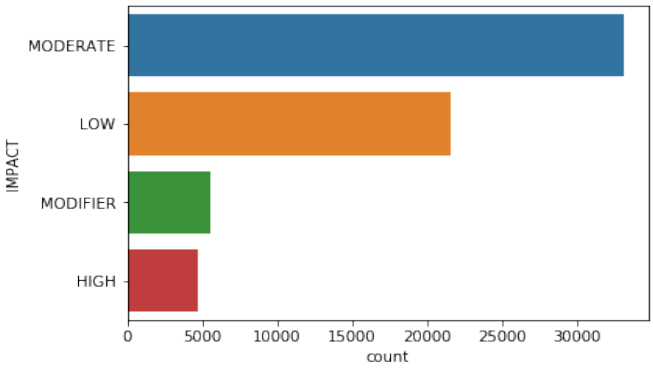


Figure 3: Bar plot of Exon_pos

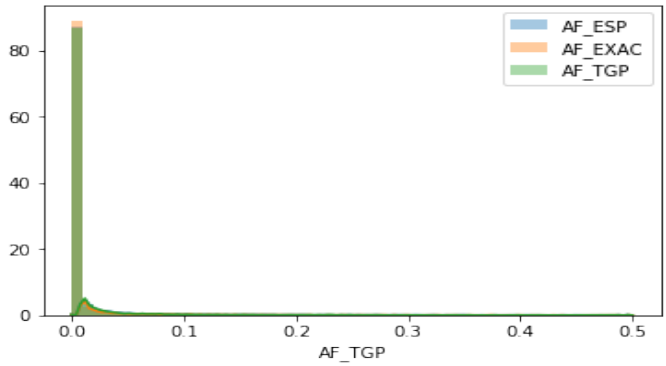
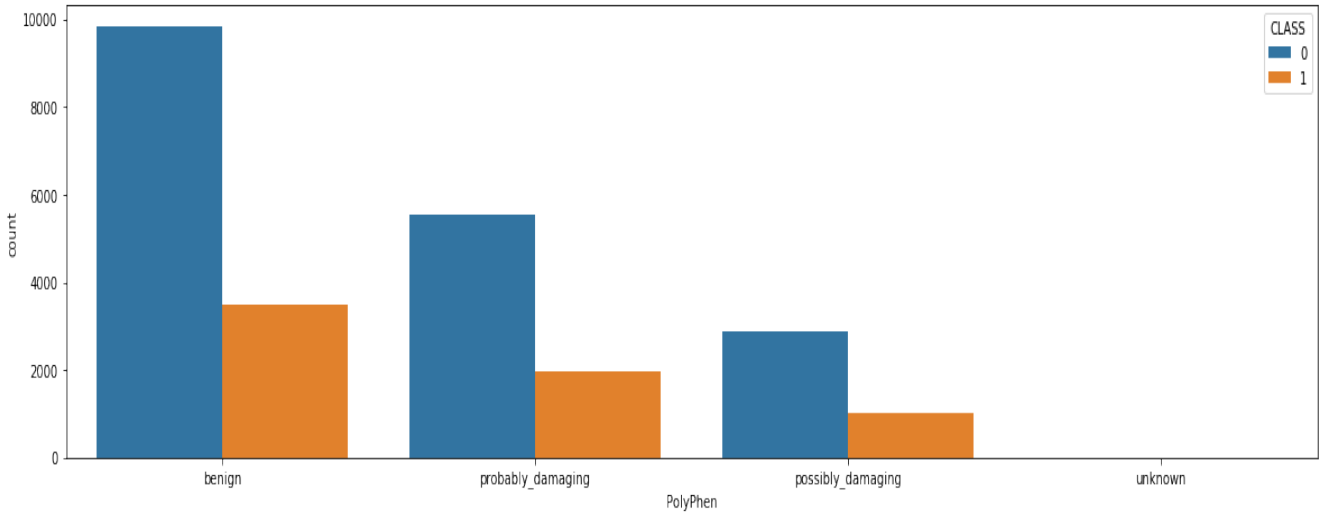


Figure 4: Graph of AF_TGP

Figure 5 : Different Bar plot of Polyphen and Amino_acids



- **Plot Numerical Distributions :**

For numerical variables we have used histograms as shown below . In below histogram we cant observe much outliers and it may sometimes not be possible to determine if an outlying point is bad . Outliers may occur due to random variation or may indicate something scientifically interesting. Experimented with few box plots but couldn't find significant outliers .

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

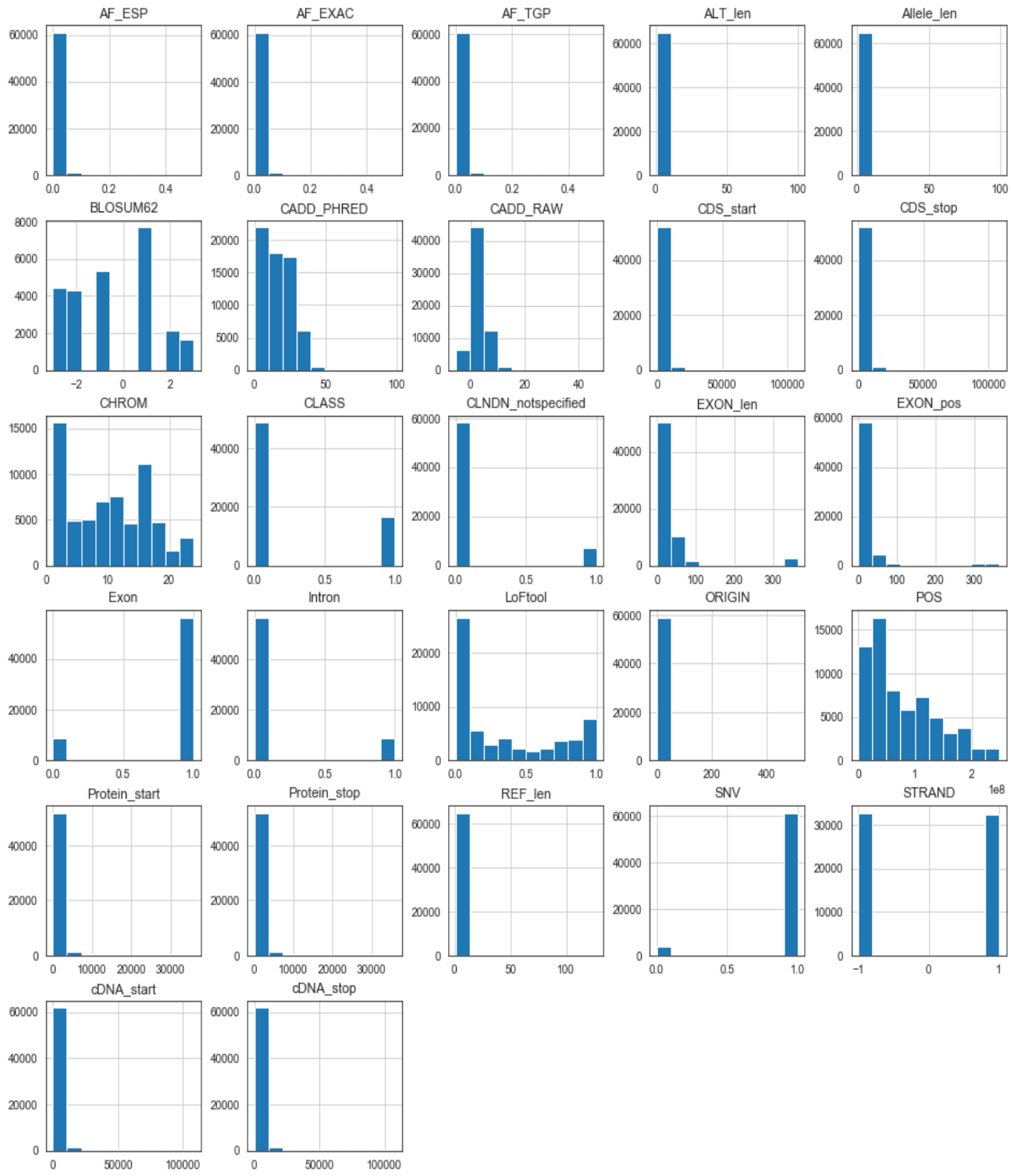


Figure 6 : Histograms of Numerical data

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

- **Correlations** : Correlation is a value between -1 and 1 that represents how two features are co-related whether they are highly correlated or not . Values near to 1 or 1 are highly correlated whereas values below 0 or near -1 are least correlated .

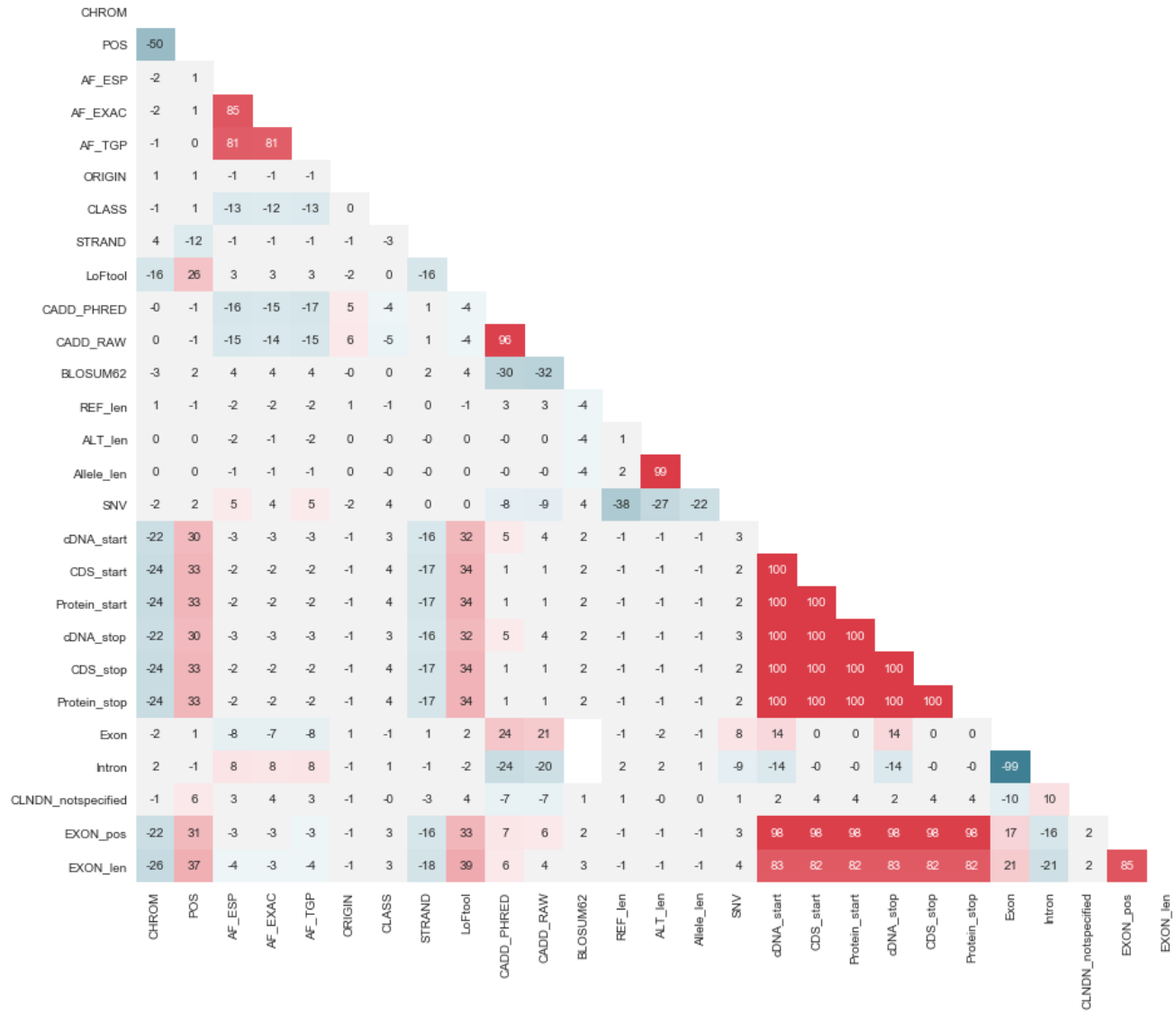


Figure 7 : Correlation plot

Feature 1	Feature 2	Correlation
CDS_stop	Protein_stop	1
Protein_stop	CDS_stop	1

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

CDS_start	Protein_start	1
Protein_start	CDS_start	1
CDS_stop	CDS_start	1

- **Github Link :** https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

STEP 5: DATA SPLIT AND BALANCE:

- Split the data into Training set, Validation set, Test set,
- Data-set is divided into 70% training set, 15% validation set and 15% test set so as to train the training set with machine-learning model and later on implement the best model in validation and test set.
- The 'CLASS' variable has imbalanced data we need to balance our training data set, it has 48754 0's and 16434 1's in our target attribute
- Balancing the data-set to equal ratio of 0's and 1's.

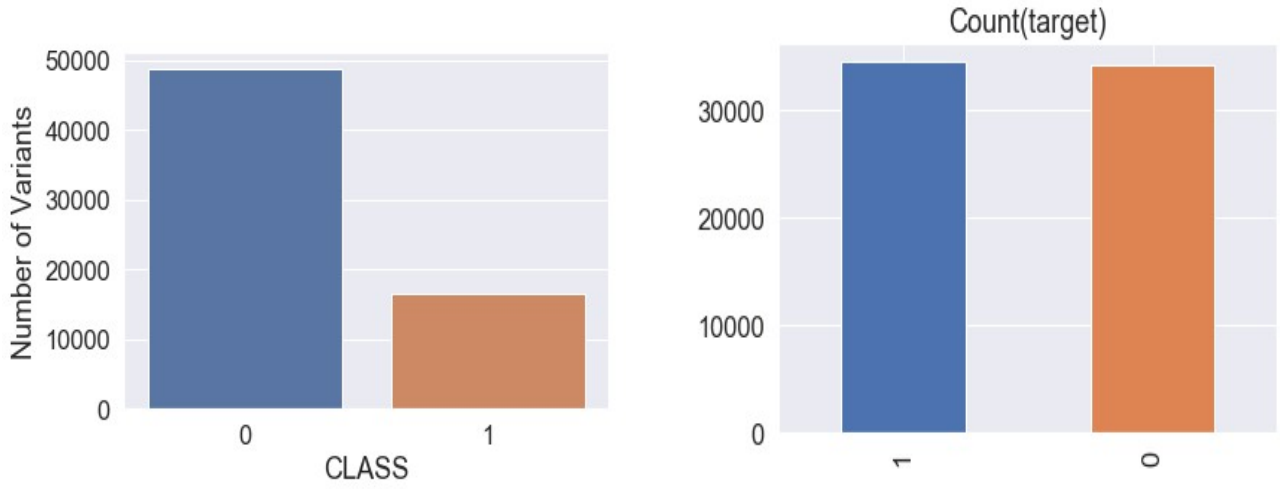


Figure 8 : Imbalanced and balanced count of consistent and conflicting classification

STEP 6: CUSTOM DATA TRANSFOMERS AND PIPELINE:

- By using 'one-hot-dense' and 'ordinal method encode categorical features as a numeric array. The input to this transformer should be a matrix of integers or strings, denoting the values taken on by categorical (discrete) features.

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

- Import CountVectorizer from sklearn.feature_extraction.text to convert a collection of text documents in 'Consequence' attribute to a matrix of token counts and implement 'onehot-dense' to return a dense array instead of a sparse matrix.
- Implement 'ordinal' method to 'IMPACT' attribute to encode the features as ordinal integers resulting in a single column of integers per feature.
- Remove NaN values from the data-set.
- By using Pipeline function combine all the categorical data into catagorical pipeline and numeric data into numeric pipeline.
- Using FeatureUnion method combine both the pipeline and fit and transform into train data as well as transform into validation and test data.
- **Github Link :** https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

STEP 7: PARAMETER TUNING AND FEATURE SELECTION :

- Use Scikit-learn's SelectFromModel transformer to perform an automated feature selection.
- Hyper-parameter tuning relies more on experimental results and best way to determine optimal parameter is by trying different combinations of parameters and evaluation performance of each model.
- **Github Link :** https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

STEP 8: IMPLEMENT MACHINE- LEARNING MODEL :

- Use three different machine learning learning model into our training data-set and compare best fitting model among the three . Here, we are using decision tree, random forest and gradient boosting algorithm.
- Check accuracy,precision,recall,ROC,AUC of the training data-set .
- Once the best model is selected implement it in combined training and validation set data set.
- Check accuracy precision,recall,ROC,AUC of the combined data-set and compare with the training set result.

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

- Finally, check the confusion matrix of the test set and measure the True Positive Rate ad False positive rate.
- **Github Link :** https://github.com/AshleyJohn24/Project/blob/master/Final%20Code.ipynb

RESULT:

The results of the analysis are summarized below.

- The data-set is comprised of 46 attributes and 65188 rows which consists of numerical as well as categorical values. The target variable is a binary representation of whether or not a variant has a conflicting clinical classification, here '0' represents consistent classifications and '1' represents conflicting classifications.
- 30 attributes were selected for machine learning : 'POS', 'AF_ESP', 'AF_EXAC', 'AF_TGP', 'LoFtool', 'CADD_PHRED', 'CADD_RAW', 'REF_len', 'ALT_len', 'Allele_len', 'SNV', 'cDNA_start', 'CDS_start', 'Protein_start', 'cDNA_stop', 'CDS_stop', 'Protein_stop', 'Exon', 'Intron', 'CLNDN_notspecified', 'EXON_pos', 'EXON_len', 'CHROM', 'IMPACT', 'STRAND', 'BAM_EDIT', 'SIFT', 'PolyPhen','BLOSUM62', 'Consequence',' CLNVC', 'Codons', 'Amino_acids', 'CLASS'**.**
- Gradient Boosting classifier is fitted into 70% training,15%validation and 15% test set.
- The confusion matrix for classification of the test cases is shown below :

	0	1
0	5252	2061
1	686	1779

- Precision: 0.88
- Recall: 0.72
- Accuracy: 0.71
- Area under the ROC curve: 0.72

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

DISCUSSION :

In our studies we are trying to predict whether the variant has an consistent classification or conflicting classification . We have split our data-set into training, validation and test set with a split ratio of 70/30 and trained three different machine learning algorithm into the training set . As our data-set comprises of binary classification problem the three supervised machine learning algorithm that we used here is decision tree, random forest and gradient-boosting classifier . By calculating the best fit score of the three models using fit function , the best score of the models are 0.76, 0.77, 0.84 for decision tree, random forest and gradient boost classifier respectively. The area under ROC curve of the three models are as follows 0.60, 0.70, 0.72 for decision tree, random forest and gradient boost classifier respectively. This concludes that the best model among the three is gradient boost .

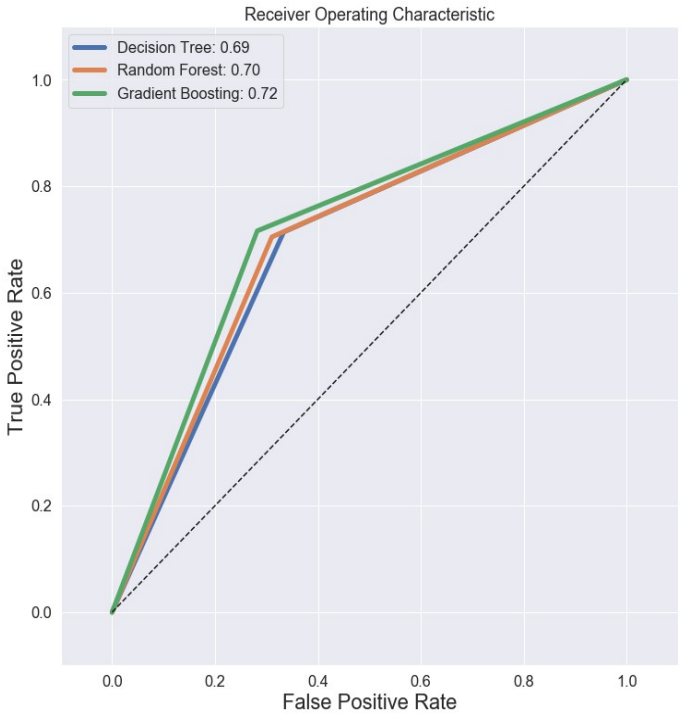


Figure 9 : ROC curve of three different models

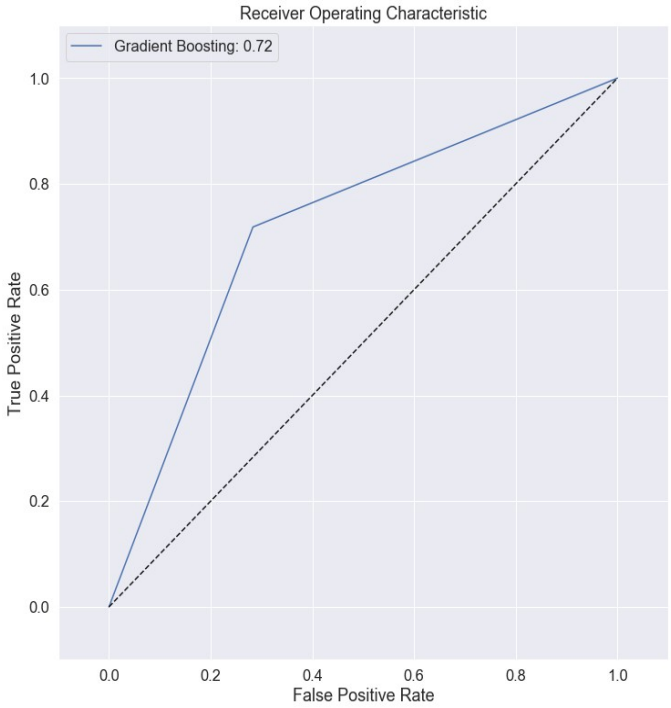


Figure 10: ROC curve of best model on test data.

Comparing result with different hyper parameters:

Sckit-learn implements a set of default parameters for all models. Best parameters are usually difficult to determine ahead of time and tuning a model is based on trial and error method . To determine an optimal parameter we have to try many different combination . But in cases , model may perform highly on the training data-set but poorly on the test data-set , which causes over-fitting . Here,

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

in our three machine-learning model we are tried using different hyper-parameters before finalizing an optimal hyper- parameters.

	Parameter 1	Parameter 2	Parameter 3
'randomforestclassifier__bootstrap'	[True]	[True]	[True]
randomforestclassifier__max_depth'	[4,5]	[8,9]	[7,8]
randomforestclassifier__max_features'	[4,5]	[8,9]	[7,8]
randomforestclassifier__min_samples_leaf	[3,4]	[7,8]	[6,7]
randomforestclassifier__min_samples_split	[8,10]	[10,12]	[10,12]
randomforestclassifier__n_estimators'	[100]	[200]	[100]
gradientboostingclassifier__max_depth	[4,5]	[8,9]	[7,8]
gradientboostingclassifier__max_features	[4,5]	[8,9]	[7,8]
gradientboostingclassifier__min_samples_leaf	[3,4]	[7,8]	[6,7]
gradientboostingclassifier__min_samples_split'	[8,10]	[10,12]	[10,12]
gradientboostingclassifier__n_estimators	[100]	[200]	[100]
decisiontreeclassifier__criterion'	['gini', 'entropy']	['gini', 'entropy']	['gini', 'entropy']
decisiontreeclassifier__max_depth'	[6,7, 8]	[9,8,10]	[7, 8,9]
decisiontreeclassifier__max_features	[6,7, 8]	[9,8,10]	[7, 8,9]
Confusion Matrix	[5048 2265] [657 1808]	[5529 1784] [823 1642]	[5252 2061] [686 1779]

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

By using different hyper-parameters as shown above, we choose optimal parameter as the 'parameter 3'. The confusion matrix obtained from 'parameter 3' provides a better recall,precision,false positive rate .

Comparing result with imbalanced and balanced data :

Applying the gradient boost classifier in both imbalanced and balanced data we obtaining the following result

- Result for imbalance data :
- Precision or Positive Predictive Value = True Positive /True Positive + False Positive = 0.80
- Recall or True Positive Rate = True Positive /True Positive + False Negative = 0.91
- Accuracy = 0.77
- Area Under the ROC curve = 0.63
- Confusion Matrix :

	0	1
0	6706	607
1	1581	884

- Result for balance data :
- Precision or Positive Predictive Value = True Positive /True Positive + False Positive = 0.88
- Recall or True Positive Rate = True Positive /True Positive + False Negative = 0.72
- Accuracy = 0.72
- Area Under the ROC curve = 0.63
- Confusion Matrix :

	0	1
0	5252	2061
1	686	1779

PREDICT WHETHER A VARIANT WILL HAVE CONFLICTING CLINICAL CLASSIFICATION

For imbalance data we mainly focus on percision and recall whereas for balance data we focus mainly on Area under ROC curve (AUC). The above results conclude that the imbalance data has highest recall and accuracy is higher than the balanced data, but as our dataset has higher rate of consistent classification variant our predictive model will be trained in such way that it will consider majority as consistent classification variant which gives higher accuracy but is deceptive , thus resulting into high performancematrix but its actually a wrong performance matrix . In such cases, we need to select performance matrix that can evaluate consistent classification and conflicting classification variant fairly. For this we have to balance that and resample the target variable with equal ratio of consistent classification and conflicting classification variant and with help of area under the ROC curve we can determine how good the model but setting a probability which is the threshold . By balancing the data we obtain AUC as 0.72 which is greater than the AUC of the imbalanced data.

CONCLUSION :

The final model yields an average Recall of 0.78, and a Recall of the conflicting class of 0.73. The average measure of test accuracy which is denoted as F1 score is 0.74 .This means that out of the 2465 genetic variants with conflicting assessment in the test set our final model captures about 1780 genetic variant with conflicting classification . By implementing this machine learning model in genetic variant data-set we anticipate that this finding might help researchers and clinicians in identifying conflicting assessments of genetic variants. Further feature selection steps based on better domain knowledge might lead to additional performance improvements. To further improve the prediction of genetic variants, one might try Artificial Neural Networks, and Deep Learning.